Pact Testing for Front End Developers

Presented by Dominick Lombardo
Software Engineer @ Uptake

Testing your code is REALLY IMPORTANT

Not
glamorous
but
totally
worth it

JavaScript Jeopardy

In 1996, the creators of JavaScript decided to find an official body that could put a standardization around the quickly-growing language.

The group they found was called ECMA, and the first standard was released in June 1997.

This group still governs the standard and their name is reflected in the official name ECMAScript 6 or ES2015.

What does ECMA stand for?

European Computer Manufacturers Association

What problem does Pact Testing solve?

JS Pitfalls

Complex infrastructures

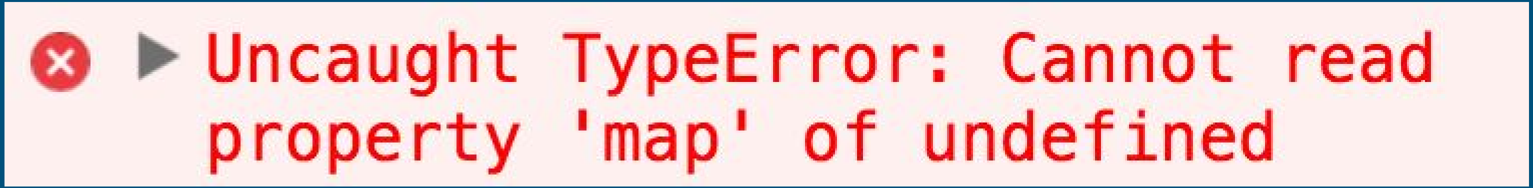
New tools all time time

Async Programming

State Management

PLUS the language itself

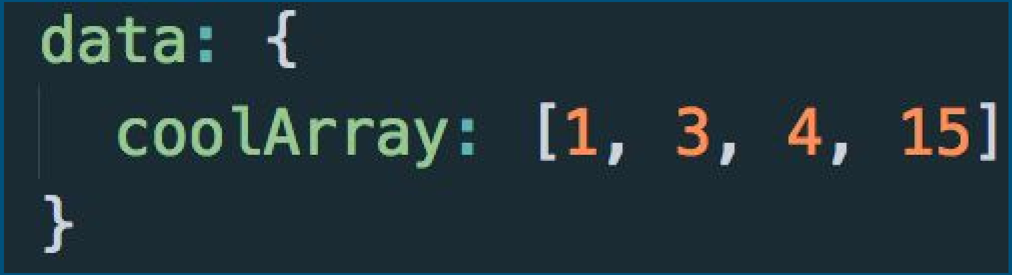
[1, 5, 3, 14].sort( ) = WTF ?



```
Uncaught TypeError: Cannot read property 'map' of undefined
```

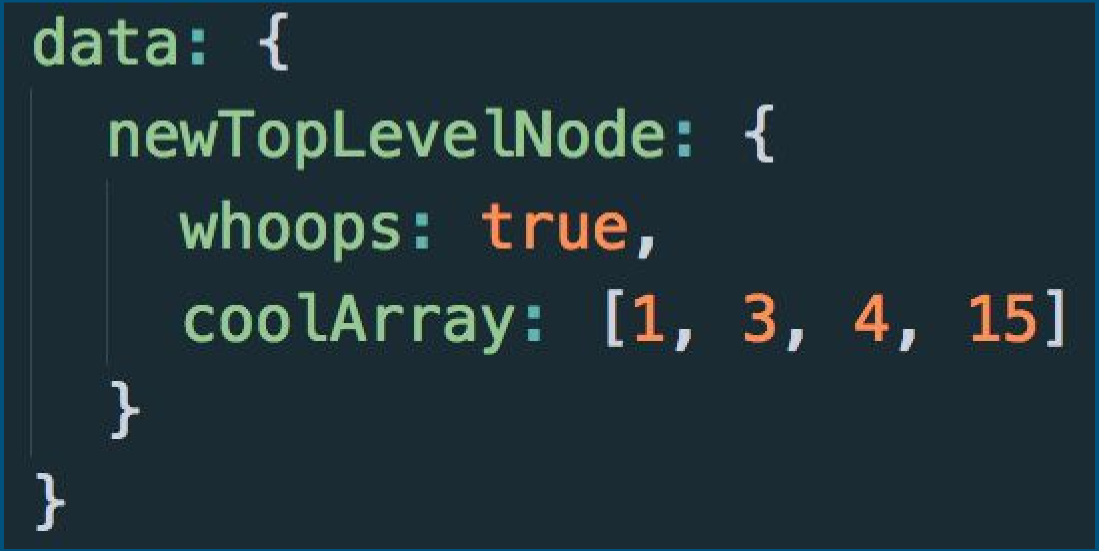
😢

Bad API!



```
data: {
  coolArray: [1, 3, 4, 15]
}
```

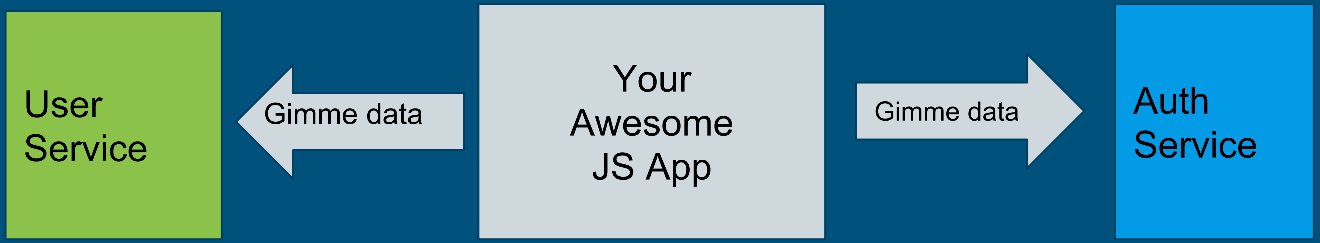
Go To
Your Room!



```
data: {
  newTopLevelNode: {
    whoops: true,
    coolArray: [1, 3, 4, 15]
  }
}
```

External Dependency Management

Managing relationships with backend APIs is the problem Pact Testing solves.



Testing Approaches

Unit Testing

Fast, but confined to application code.

Integration Testing

Thorough, but be expensive in terms of resources and time.

The Best Of Both

Pact Tests are written like unit tests, but perform like integration tests.

Pact

noun

compact; an agreement or covenant between two or more parties

Our Definition

A contract between a front-end application and a back-end application that defines the interactions between the two systems

application

other application

Informal Pacts

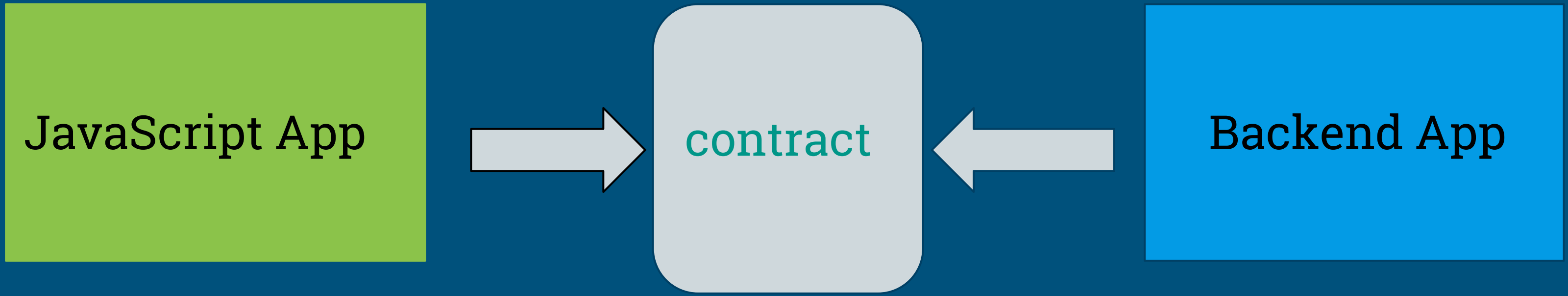
You may be guilty

Checking out the readme

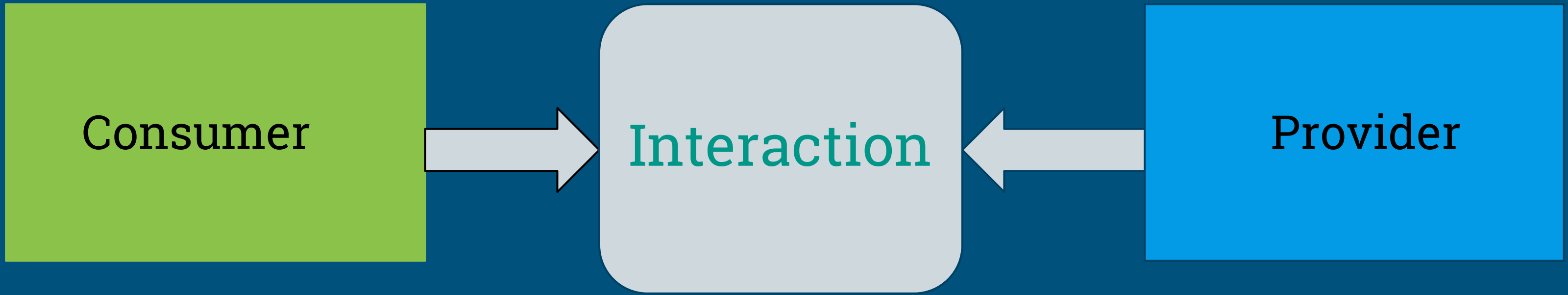
Playing with Swagger UI

Discussions with backend team

High Level Overview



Pact Lingo

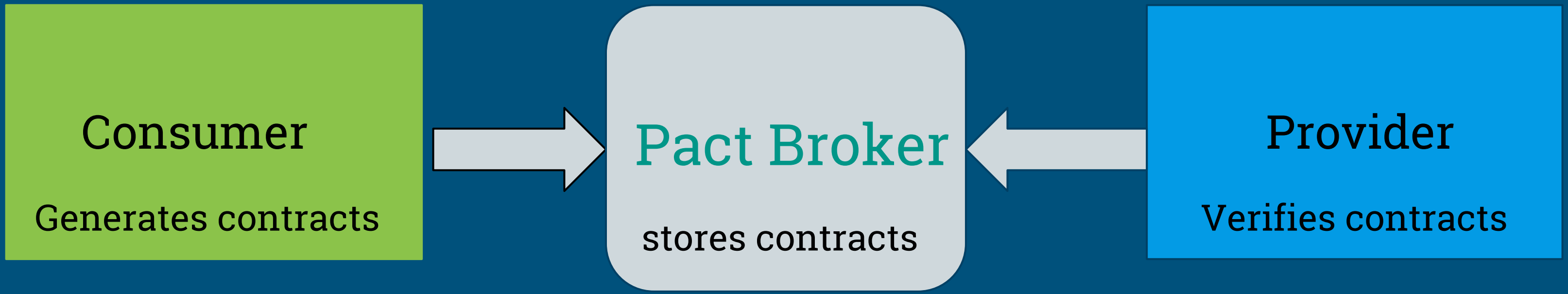

Consumer

Interaction

Provider

So how's it work?

JSON contracts



Consumer

Generates contracts

Pact Broker

stores contracts

Provider

Verifies contracts

Quick Review

are you still awake?

Dependency Management

Contracts

Generated by frontend devs

Verified by backend devs

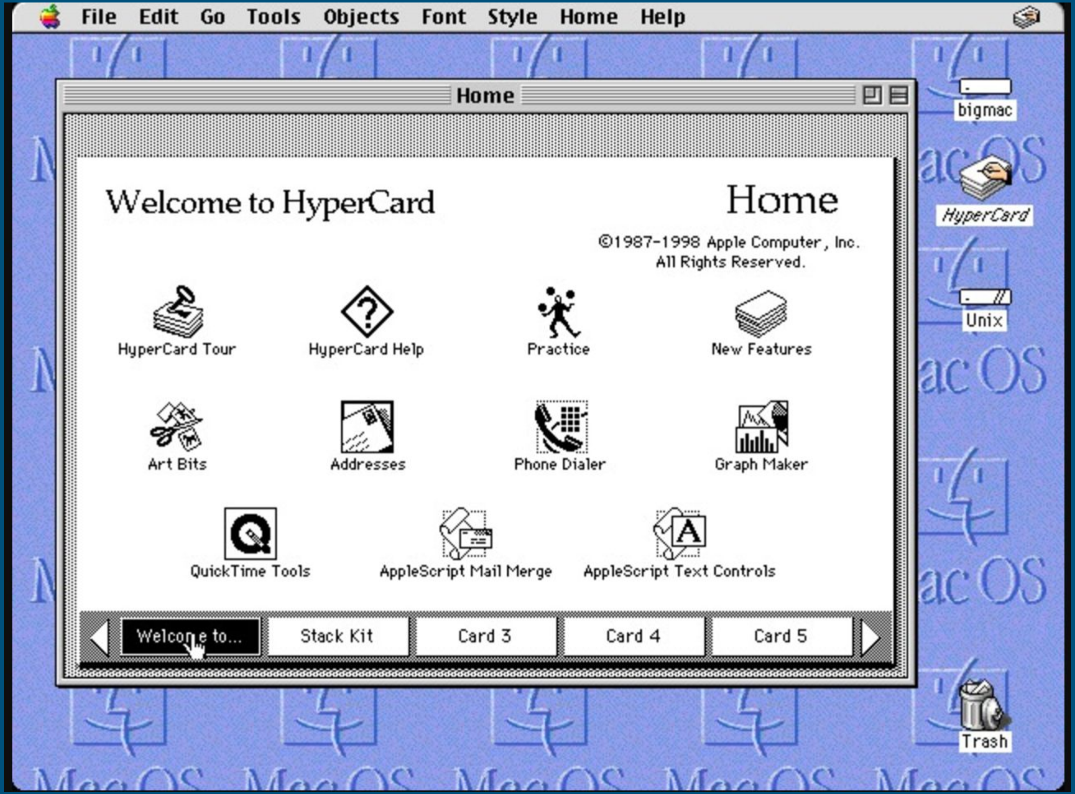
JavaScript Double Jeopardy

There was an application, introduced by Apple in 1987, that inspired Netscape's Brendan Eich as he created JavaScript.

This application allowed event-driven programming thru the use of event handlers. GUI elements, such as buttons and fields, could be added to the UI and react to user input.

What is the name of this application?

HyperCard



Welcome to HyperCard

Home

©1987-1998 Apple Computer, Inc.
All Rights Reserved.

Testing Pattern

Given...

When...

Then.

Pact’s Version of Given-When-Then

GIVEN a provider state

WHEN a certain HTTP Request is made

THEN a specific HTTP response is expected

HTTP Request/Response Ingredients

Request

HTTP Verb *get, post, put, delete*

Requested Resource */path/to/cool/stuff*

Headers

Request Body (optional)

Response

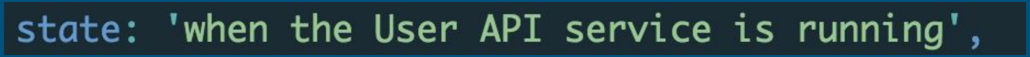
Status Code *200, 409, 503*

Headers

Response Body (optional)

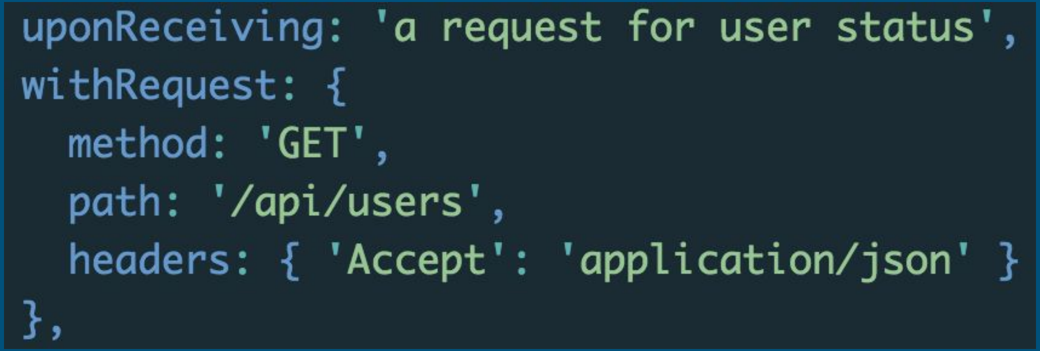
Pact = Given/When/Then + HTTP

Given



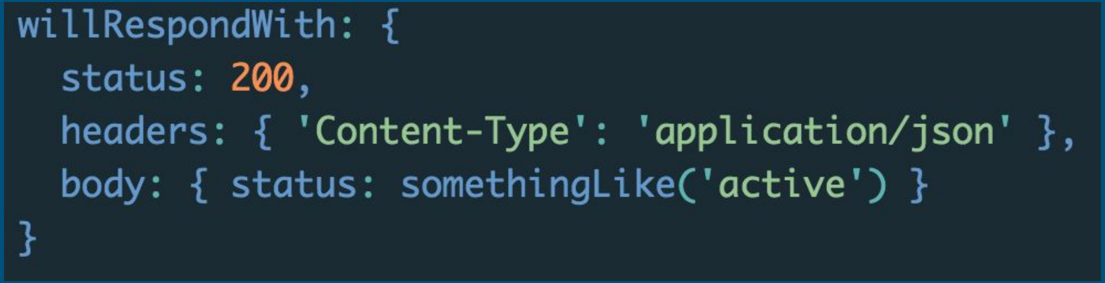
```
state: 'when the User API service is running',
```

When



```
uponReceiving: 'a request for user status',
withRequest: {
  method: 'GET',
  path: '/api/users',
  headers: { 'Accept': 'application/json' }
},
```

Then



```
willRespondWith: {
  status: 200,
  headers: { 'Content-Type': 'application/json' },
  body: { status: somethingLike('active') }
}
```

Warning

Pact Is the Wild West Of Testing Frameworks

Not easy to implement

Expensive in terms of time

I hope you like reading source code

Live Demo

Thanks For Listening

email:

dpa.lombardo@gmail.com

github (talk slides & code posted):

@ lombardo-chcg

5 times per week blog posts:

https://lombardo-chcg.github.io